PART-A

L.VALLIAPPAN
39111054
193111355

1.) write the R script to add, subrat, multiply two matrices

```
# creating first matrix
a <- matrix(data = 1:9 , nrow= 3, ncol=3)

# creating second matrix
b <- matrix ( data = 1:9, nrow =3, ncol=3)

print(" First matrix")
print (a)
print(" second matrix")
print (b)
# Add a and b
c <- a+b
print (" Adding")
print(c)
# sub a and b
d = a-b
.print (" subtract")
print(c)
# multiply a and b
e = a *b
print (" multiply")
print (e)
```

2.) Differentiate break and next in R with example.

Break =

program:

```
x2 <- 1:5
for (val in x2) {
    if (val == 3) {
        break
    }
    print (val)
}
```

output

```
1
2
```

L.VALLIAPPAN
39111054
19311355

we are assigning x2 as from 1, 2, 3, 4, 5.
when in for loop it reaches Val as 3 as per condition it break the entire loop and stop forther execution.

Next -

```
x2 <- 1:5
for(val in x2) {
   if (val == 3) {
      next
   }
   print(val)
}
```

OUTPUT

```
1
2
4
5
```

As same value of x2 is taken ~~so~~ and same if condition. here if loop get val == 3 it skip that execution ~~and~~ alone and print rest all.

~~3)~~

4.) find mean, median, ~~mode~~ standard deviation.

```
A = (2, 4, 9, 10, 14, 28, 52)
print (" mean")
print (mean (A)).
print ("median")
print(median(A)).
print (" standarad deviation")
 d <- Sqrt (var(A))
print (d)
```

L.VALLIAPPAN
39111054
19S111355

PART-B

5.) Explain control structure in R with Example

control structures =

* If condition
* If- else condition
* for loop
* nested loops.
* while loops
* repeat and break statement
* return Statement
* next Statement

-> If condition =
This control strol structure checks the expression provided in parenthesis is true or not. If true, the execution of the statements in braces {} continues

example=

```
x <- 100
if (x > 10) {
  print (paste (x, "is greater than 10"))

}
```

output:

```
  100 is greater than 10.
```

-> If - Else =
It is similar to if condition but when the test expression in if condition, then statement in else condition are executed

Example =

```
name = readline (prompt = "ENTER YOUR NAME:")
age = as. integer (readline (prompt = " ENTER YOUR AGE:"))
print (& " AGE VERIFICATION FOR VOTE")
print (Paste (" YOUR NAME IS :", name))
print (Paste (" YOUR AGE IS : ", age))
if (age>18) {
    print (" YOU ARE ELIGIBLE TO VOTE")
}else {
    print (" YOU ARE ELIGIBLE TO VOTE")
 }
```

L-VALLIAPPAN
39111054
19S111355

OUTPUT

```
ENTER YOUR NAME : VALLIAPPAN
ENTER YOUR AGE : 19
"AGE VERIFICATION FOR VOTE"
"YOUR NAME IS : VALLIAPPAN"
"YOUR AGE IS : 19"
~~"YOUR AGE IS~~
"YOU ARE ELIGIBLE TO VOTE"
```

-> FOR LOOP

It is a type of Loop (or) sequence of statements executed repeatedly until exit condition is reached.

example

```
x <- c (-8, 9, 11, 45)
for (i in x)
{
  print (i)
}
```

OUTPUT

```
-8
9
11
45
```

-> NESTED LOOP

It is similar to simple loops. Nested means loop. More over, nested loops are used to manipulate the matrix.

example

```
m <- matrix (2:15, 2)
for (r in seq (nrow (m))) {
  for (c in seq (ncol (m))) {
     print (m[r, c])
  }
}
```

OUTPUT

```
2            9
4            11
6            13
8            15
10
12
14
3
5
7
```

L-VALLIAPPAN
39111054
19SU1355

-> while loop :

It is another kind of loop iterated until a condition is satisfied. The testing expression is first before executing the body of loop.

Example

```
x=1
while (x<=5) {
    print (x)
     x=x+1
  }
```

OUTPUT

```
1
2
3
4
5
```

-> repeat loop and break statements.

It is a loop which can be iterated many number of times but there is no exit condition to come out from the loop. So, break statement is used to exit from the loop.

Break statement can be used in any type of loop to exit from the loop

Example -

```
x=1
repeat {
    print (x)
     x=x+1
     if (x>5){
          break
     }
 }
```

OUTPUT

```
1
2
3
4
5
```

-> Return statement

It is used to return the result of an executed function and return control to the calling function

Example

```
func <- function(x){
 if (x>0){
      return("positive")
 }else if (x<0){
      return("Negative")
}else{
```

L.VALLIAPPAN
39111054
19311355

```
    retorn("zero")
  }
}
fun(1)
fun(0)
fun(-1)
```

OUTPUT =

```
"positive"
"zero"
"Negative"
```

⇒ Next Statement

It is used to skip the current iteration without executing the further statements and continues the next iteration cycle without terminating the loop.

Example:

```
x <- 1:10
for (i in x) {
    if (i %% 2! = 0) {
        next
    }
    print (i)
}
```

OUTPUT

```
2
4
6
8
10.
```

L.VALLIAPPAN
39111054
195111355

8) Data visualization

It is use to express the charts of the printed code of the R program.

- Bar chart
- histogram
- Box plot
- scatter plot
- Heat map

Bart chart =
syntax (H, x, lab, ylab, main,

histogram = to see count of the numberr

hist (main, xlab, xlim, ylim)

Box plot
= It depicts five statiscall numbers.
boxplot (x, data, notch, names)

calculate mean

Scatter plot =
It depicts number points in the graph

syntax
scatter plot (x, data, xlab, ylab, xlim, ylim)

L.VALLIAPPAN
39111054
19511355

Heat map=

= graphical representation of data in different coloors.

L.VALLIAPPAN
39111054
19S111355

9.)

a.) Linear regression

~~those~~

It is a very widely used statistical tool to establish a relationship model between two variables - one of these variable is called a predictor variable whose value is gathered through experiments. The othe variable is ~~called~~ derived from the predictor Variable.

$y = ax + b$

y = response variable

x = predictor variable

a and b are constants.

b.) Height (in cm) of Father and Son are given as follows.

Father (x)	150	152	155	157	160	161	164	165
Son (y)	154	156	158	159	160	162	161	164

program

```
x <- c( 150, 152 , 155, 157 , 160, 161, 164, 165)
# The predictor vector is above
y <- c( 154, 156 , 158, 159, 160, 162, 161, 164)
# The response vector is above

relation <- lm(y~x)
# Apply the lm() function.

# Find height of son as Father given
a <- data.frame(x = 170)
result <- predict(relation, a)
print(result)
```